*Robofest-2013*

*Held at Lawrence Technological University, Southfield, Michigan*

*On 17th-18th May, 2013*

Objective:

An autonomous robot is to search for & rescue people from a tower (trapped in a black box), collect data, and clean up a contaminated area with a tower of boxes. Detailed missions are to

1. Remove (clean up) the white toxic boxes from the table.
2. Bring the black box out of the contaminated area to Home.
3. Measure the size of the contaminated area in square millimetres and report the number.
4. Return to the Home Base.

For Sr. Division the contaminated area is a right angled triangle.

Problems Faced:

1. Value of light intensity given by light sensors varying.
2. Weight management.
3. Problems of using mechanix.
4. Box slipping from the arm.
5. Shortage of parts.
6. Accuracy of the reading to be displayed.
7. Combining to different codes.
8. Problems while using small tires.
9. Balancing with 2 wheels at the back.
10. Wire not reaching the ultrasonic sensor on the arm of the robot.

Solutions:

1. Due to the surrounding light, the light sensors used to give variable values for white and black. To stabilize this, we put cardboard around the light sensors to block the surrounding light.
2. When we attached the levels to our robot, the weight increased. This had effects on the movements of the robot. The motor couldn’t pull the robot forward. To deal with this, we organised the weight on the hand and made the levels of aluminium instead of lego and mechanix.
3. Usage of mechanix led to several problems. First of being the stability of the robot. The mechanix parts are very flexible. Therefore when the robot was moving with box, the arm was swaying a lot. Second, the mechanix added a lot of weight to the robot.
4. To hold the box better, we attached foam and rubber to it. We saw that the arm had lots of spaces here there. To cover up this space, we used foam. The rubber added friction to hold the box.
5. As we had only two lego kits, the number of parts was limited. Therefore to conserve parts, we organized the arm in a better way.
6. The tires tended to slip a lot. This leaded to an error in the reading. To minimize the error, we used a factor which was to be multiplied to the area. We learnt that there could be 2 types of errors:
7. Addition: In this, a number is to be added. This could be because the robot might not be able to go on a certain space of the triangle.
8. Ratio: This is when the area of the triangle increases or decreases proportionally.
9. We made our codes part by part. Then when we combined it together, we got error. There were many factors. One was that, when we want to check if something equal to something (example, i=1), we should write i= = 4, whereas we wrote i=1. Another factor placing of brackets in the code. As we did not indent the code we didn’t know where the brackets used to code. So we had to indent the code so that we could close the necessary brackets.
10. When we used small tires the robot used to scratch the ground. Therefore when the box is kept, the robot almost stopped moving. To solve this, we used big tires. Using this, the robot was lifted up so that it couldn’t scratch the ground.
11. Due to the weight of the robot, the wheels at the back of the robot started to slip out. As there were two wheels at the back, the wheels used to bend out. To reduce the pressure piled up on the two wheels, we thought of using 4 wheels. This ensured that the wheels behind could take the weight.
12. The ultrasonic sensor was attached to the arm in the beginning. If the arm was at the 5th level, the wire couldn’t reach the ultrasonic. To deal with this, we made attached the ultrasonic at the base.